



# Musketeer: Incentive-Compatible Rebalancing for Payment Channel Networks

Zeta Avarikioti  

TU Wien and Common Prefix, Austria

Stefan Schmid  

TU Berlin, Fraunhofer SIT, and Weizenbaum Institute, Germany

Samarth Tiwari  

Centrum Wiskunde & Informatica, Netherlands

## Abstract

In this work, we revisit the severely limited throughput problem of cryptocurrencies and propose a novel rebalancing approach for Payment Channel Networks (PCNs). PCNs are a popular solution for increasing the blockchain throughput, however, their benefit depends on the overall users' liquidity. Rebalancing mechanisms are the state-of-the-art approach to maintaining high liquidity in PCNs. However, existing opt-in rebalancing mechanisms exclude users that may assist in rebalancing for small service fees, leading to suboptimal solutions and under-utilization of the PCNs' bounded liquidity.

We introduce the first rebalancing approach for PCNs that includes *all users*, following a “*all for one and one for all*” design philosophy that yields optimal throughput. The proposed approach introduces a double-auction rebalancing problem, which we term MUSKETEER, where users can participate as buyers (paying fees to rebalance) or sellers (charging fees to route transactions). The desired properties tailored to the unique characteristics of PCNs are formally defined, including the novel game-theoretic property of *cyclic budget balance* that is a stronger variation of strong budget balance.

Basic results derived from auction theory, including an impossibility and multiple mechanisms that either achieve all desiderata under a relaxed model or sacrifice one of the properties, are presented. We also propose a novel mechanism that leverages time delays as an additional cost to users. This mechanism is provably truthful, cyclic budget balanced, individually rational and economic efficient but only with respect to liquidity.

**2012 ACM Subject Classification** Computing methodologies → Distributed algorithms; Theory of computation → Distributed algorithms; Theory of computation → Algorithmic mechanism design

**Keywords and phrases** Blockchains, Payment Channel Networks, Rebalancing, Game Theory

**Digital Object Identifier** 10.4230/LIPIcs.AFT.2024.8

**Funding** This work was partially funded by the German Research Foundation (DFG) Schwerpunktprogramm (SPP 2378, ReNO) 2023-2027, by the Austrian Science Fund (FWF) through the SFB SpyCode project F8512-N, the project CoRaF (grant agreement ESP 68-N), and by the WWTF through the project 10.47379/ICT22045.

## 1 Introduction

### 1.1 Motivation

Bitcoin and other cryptocurrencies are significantly transforming the financial landscape [35, 50]. However, a well-known issue of the celebrated Nakamoto consensus introduced with Bitcoin, is that it inherently prohibits high transaction throughput which in turn hinders the widespread adoption of blockchain technologies [18]. For example, Bitcoin can process at most 7 transactions per second [18], while Visa processes tens of thousands of transactions



© Z. Avarikioti, S. Schmid, S. Tiwari;

licensed under Creative Commons License CC-BY 4.0

6th Conference on Advances in Financial Technologies (AFT 2024).

Editors: Rainer Böhme and Lucianna Kiffer; Article No. 8; pp. 8:1–8:22

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

per second. Furthermore, blockchains are evidently environments for-profit, therefore user-incentive design is critical. Although several works have studied blockchain-related topics under the lens of game theory, e.g., [13, 21, 16, 27, 14], there is still much to be explored, particularly concerning scaling protocols. *In this work, we model and investigate incentive-compatible mechanisms that can enhance the limited transaction throughput of blockchains like Bitcoin.*

Specifically, we focus on one of the most prominent and well-studied scalability solutions for blockchains, called *payment channels* [38]. With payment channels, users can transact off-chain at far lower costs and faster speeds. The core idea is that any two users can lock their coins in a “joint account” on-chain, namely the payment channel. Thereby, the channel parties may perform arbitrarily many off-chain transactions with each other by signing messages with the new distribution of coins in their joint account. To close the payment channel, the parties can publish on-chain the last update on the distribution of their coins. Naturally, each channel is limited by the coins locked by each party (liquidity), dictating the maximum amount that can be sent between them. For example, in a channel with Alice and Bob currently holding 3 and 5 coins respectively, Alice can send at most 3 coins to Bob, and Bob can send at most 5 coins to Alice. In short, the coins can be moved on the channel from Alice to Bob or vice versa, much like moving balls from one side of an abacus to the other.

Multiple payment channels operating on the same underlying blockchain, comprise a *payment channel network (PCN)*. PCNs allow users, who have at least one payment channel open, to route transactions through the network to other users with whom they do not share a direct payment channel. To successfully route a transaction, a path of channels with sufficient liquidity for all senders must exist. For example, if Alice wants to send 3 coins to Carol through Bob, Alice must have 3 coins available in her channel with Bob, and Bob must have 3 coins available in his channel with Carol. The intermediaries (e.g., Bob) that offer to use their channel liquidity to route another user’s transaction typically ask for a routing service fee. If a channel in the selected path is depleted (i.e., has low liquidity) in the desired direction, all the transfers in the path will be reverted and the transaction will fail. *The liquidity of individual payment channels is, therefore, a crucial factor in the effectiveness of PCNs as a scaling solution.* It determines the ability to route transactions and impacts the overall efficacy of PCNs in enhancing the transaction throughput.

To maintain high liquidity in PCNs, parties have two options: either lock a significant amount of coins initially or use an on-chain transaction to top up their channels. However, both options have their drawbacks. Locking a substantial amount of coins incurs an opportunity cost as these coins cannot be used for other on-chain operations. On the other hand, using on-chain transactions to top up channels hinders the scaling capabilities of the underlying blockchain.

*Rebalancing mechanisms* are an attractive alternative solution to improve liquidity within PCNs [26, 10, 1]. These mechanisms aim to identify cycles of depleted edges (channels) and route transactions across them in a way that ensures each node in the network has an equal amount of coins at the end of the process. By leveraging cycles within the PCN, parties with depleted channels can rebalance their channels by utilizing two of their channels – one as a source to send coins and another as a destination to receive coins.

However, the deployed local rebalancing algorithms [1] may be practically insufficient for two main reasons. Firstly, they only involve parties interested in rebalancing, thereby excluding channels that may route transactions for low or no routing fees; after all, intermediaries are indifferent to whether the routed payment concerns a payment (path) or rebalancing

(cycle). Secondly, local searching algorithms may miss optimization opportunities leading to poor outcomes.

To address the latter limitation, Revive [26] proposed globally coordinated channel rebalancing, therefore, achieving optimal outcomes. Hide & Seek [10] recently improved on Revive by enabling global rebalancing in a decentralized and privacy-preserving manner. However, in both algorithms, the rebalancing subgraph only includes the parties that wish to rebalance while the vast majority of channels of the PCN that may route transactions for low or no fees are neglected. *Thus, even with globally coordinated rebalancing, the limited rebalancing subgraph still impacts the optimality of the overall solution, and subsequently the PCN's scaling capability, i.e., how many transactions can succeed off-chain given a bounded overall liquidity.*

## 1.2 Our Contribution

We propose a novel approach to rebalancing that *involves all PCN users* in order to maximize the liquidity utilization and subsequently the transaction throughput. Our approach allows all users to submit their liquidity and bid for every one of their channels. The liquidity in this setting captures the number of coins they are willing to use for routing/rebalancing while the bid encapsulates how much they are willing to pay per coin for rebalancing the specific channel. So positive bids express the desire of buyers to rebalance, whereas negative (and zero) bids the desire of sellers to sell their routing service. Now, modeling this problem reveals a major challenge: *how can we design an incentive-compatible rebalancing mechanism for both buyers and sellers?*

To the best of our knowledge, we are the first to examine user incentives in the context of rebalancing mechanisms for PCNs. Our goal is twofold: First, to formally model the problem, capturing the unique characteristics present in PCNs; second, to discover satisfactory solutions, exploring different trade-offs. To achieve our objectives, we extend Hide & Seek [10] to accommodate both buyers and sellers of rebalancing liquidity. This approach leads to a double-auction problem with several challenges stemming either from traditional auction theory or from the individual needs of PCNs. In modeling our problem, we pinpoint *channel depletion* as a distinct feature, setting it apart from other network mechanism designs like routing games [22]. Channel depletion signifies that transactions can *permanently* lower an edge's capacity (here, liquidity) until counteracted by an opposite flow. Unlike railway networks where trains need tracks only temporarily, flows in our model can *compensate* for each other. Thus, existing results do not directly apply.

To determine the desiderata of our mechanism, we revisit conventional requirements from auction theory: (1) *economic efficiency*, i.e., maximizing the social welfare which captures that channels are prioritized for rebalancing based on their bids, (2) *truthfulness*, meaning users submit their true value, and (3) *individual rationality*, i.e., non-negative utility for rebalancing participants. However, our problem encounters an idiosyncrasy rooted in the payment channel primitive itself, affecting the budget-balancedness of the mechanism, i.e., the mechanism does not incur a deficit (nor a surplus). Specifically, coins cannot be burned in a payment channel because intuitively channel updates must always benefit one party; if there exists a coin distribution where both parties in the channel can benefit from changing, then there is no way to enforce it. For instance, we cannot enforce a distribution of 3 coins to Alice and Bob each and 2 coins burned, because the parties will cooperatively update their channel to hold 4 coins each. This implies that the mechanism cannot have either a surplus or a deficit, rendering (weakly) budget-balanced mechanisms infeasible. What's more, rebalancing itself occurs via individual cycles in the PCN. As a result, our setting demands a

stronger notion of budget balance, which we term (4) *cyclic budget balance*, i.e., each cycle must be strongly budget balanced independently.

Unfortunately, the above four desired properties cannot be simultaneously achieved by any mechanism. We prove this by applying the classic Myerson-Satterthwaite impossibility result for double auctions [33]. We further emphasize the significance of the cyclic budget balance property in shaping potential solutions: The output of a rebalancing mechanism consists of a set of rebalancing circulations, which are global solutions where user preferences in one segment of the graph can impact the rebalancing cycles in distant segments of the graph. While in VCG-type mechanisms users are compensated for the global effects of their channels, the constraint of cyclic budget balance prevents this approach.

To provide satisfactory solutions, we apply standard techniques such as the renowned VCG mechanism and first-price auctions to the problem at hand. In particular, we showcase a mechanism that satisfies all the desired properties but is only applicable when all users are aware of the potential maximum and minimum fees they might pay or earn for their participation. Subsequently, we present a VCG-type mechanism that also satisfies all the desiderata exclusively for buyers, under the assumption that sellers are not treated as strategic agents. We then provide a mechanism that also considers sellers but, similarly to first-price auctions, sacrifices truthfulness. Finally, we propose a novel mechanism that introduces *time delays* as a natural characteristic of this problem, with the aim of incentivizing users to actively and truthfully participate in the rebalancing process while optimizing the outcome. The inclusion of time delays allows us to navigate around the impossibility and maintain our objective of maximizing rebalanced liquidity, in exchange for losing economic efficiency in terms of time delays and liquidity combined.

## 2 Preliminaries and Model

In this section, we first provide the necessary background on the rebalancing of payment channel networks, which we subsequently use to introduce our setting and problem definition, termed MUSKETEER. We further present an overview of MUSKETEER. For the rest of the paper, we use the terms users and players interchangeably.

### 2.1 Rebalancing PCNs

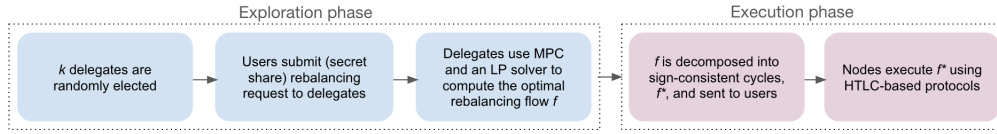
Rebalancing mechanisms are currently the only approach that allows users to restore their channel balances off-chain. In a nutshell, rebalancing mechanisms search the payment network for depleted channels that users wish to top-up off-chain until they identify a cycle of channels with enough liquidity. For instance, suppose Alice has one depleted channel with Bob, which she wants to top-up for 3 coins, and another channel with Carol where she has plenty of coins. Now, if Bob and Carol share a channel with at least 3 coins available for Carol, Alice can send 3 coins to Carol in their channel, Carol 3 coins to Bob, and Bob 3 coins to Alice. This way all users end up with the same total amount of coins. We stress that coins locked in a channel cannot be transferred to any other channel, much like the balls in different rows of an abacus.

Rebalancing mechanisms fall into two categories: local and global. Local rebalancing, currently deployed on the Lightning Network [38], has each party searching individually the network for other channels that want rebalancing; if a cycle is identified then the party can rebalance its channel. This approach may not find the optimal solution for rebalancing and it is very inefficient. Global rebalancing, introduced with Revive [26] and subsequently optimized with Hide & Seek [10], finds the globally optimal solution for the users that directly

and personally benefit from rebalancing their channels by leveraging coordination. Our solution extends this approach to further include users who are indifferent to rebalancing or may be willing to participate for a very small service routing fee. Considering routing fees are orders of magnitude smaller than the typical fee paid to the blockchain to top-up the channel balance, it is cost-effective for users to pay intermediaries to facilitate their rebalancing, similarly to transaction routing in PCNs – instead of paths, they route in cycles. We detail below the Hide & Seek mechanism that underpins our solutions.

## Hide & Seek [10]

The protocol proceeds in two phases: the exploration phase which identifies rebalancing cycles, and the execution phase which ensures their atomic execution in a secure and incentive-compatible fashion. The exploration phase begins with the random selection of  $k$  delegates among the users, e.g., using cryptographic sortition [23]. Then, the users submit their rebalancing requests, i.e., how many coins they wish to rebalance, to the delegates using secret sharing. Thereafter, the delegates use multi-party computation to calculate the optimal rebalancing flow on the network. To preserve users' privacy, each user receives only their specific flow. The optimization problem is modeled as a linear program that maximizes the rebalancing flow. The execution phase initiates by decomposing this flow into simple sign-consistent cycles, meaning that each channel only shares cycles with flow in that same direction. As a result, the channel owners are incentivized to execute all channels, and not select a subset thereof. The execution of the cycles occurs atomically, i.e., either all transactions succeed or all fail, using HTLC-based solutions [38, 47, 49]. Figure 1 illustrates the Hide & Seek protocol flow.



■ **Figure 1** The protocol flow of Hide & Seek.

## 2.2 Musketeer Overview

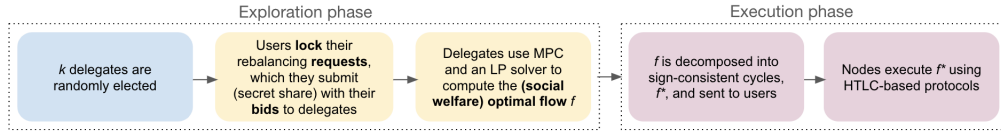
In MUSKETEER, each PCN channel may participate in the rebalancing process either as a depleted or as an indifferent edge. Depleted edges are channels owned by players that wish to rebalance their channels (i.e., act as *buyers*), while indifferent edges are owned by players that sell their routing services (i.e., act as *sellers*). We model this problem as a double auction: each player submits their (non-negative or non-positive) bid for each channel they are part of, which indicates the maximum or minimum amount they are willing to pay or receive per unit coin for rebalancing or routing through that channel, respectively.

Additionally, for each channel, the users submit their liquidity, i.e., the number of coins available to the rebalancing mechanism. These coins may be available because buyers want to rebalance their channels or because sellers may want to earn fees for their service. With this knowledge, we extract the rebalancing subgraph, which is a directed graph with capacities capturing each channel's liquidity.

The resulting combinatorial problem can be modeled as a max-flow problem, where the goal is to maximize the total number of coins (flow) weighted by the buyer's bids. In other words, we calculate the flow that maximizes social welfare, respecting the channel

capacities. We then decompose the flow in simple independent cycles that may be executed atomically [10]. Our main problem is pricing each cycle separately, awarding fees to sellers paid by the buyers.

MUSKETEER's participants are required to pre-lock the coins intended for rebalancing prior to the mechanism revealing the individual cycles. This design decision is primarily to prevent buyers from choosing whether to proceed with rebalancing after the output of the mechanism is known, as this could potentially incentivize dishonest strategies. From a different perspective, if buyers have the option to abort the mechanism in hindsight, the effectiveness of the mechanism may be severely hindered as a cycle can only be executed only if all players choose to participate and lock their coins. Figure 2 illustrates MUSKETEER integrated into the Hide & Seek protocol flow.



■ **Figure 2** The backbone of MUSKETEER, integrated into the Hide & Seek protocol flow.

## 2.3 Model and Notation

### 2.3.1 Payment Channel Network (PCN)

A payment channel network can be modeled as an undirected graph, with a vertex for every user and an edge connecting users  $u, v$  whenever they jointly own a payment channel, as depicted in Figure 3(a). At any point in time, the (bidirectional) capacities of the payment channel and its current distribution of coins can be encoded as follows: the capacity of edge  $e = (u, v)$  in the direction from  $u$  to  $v$  is the maximum amount of money that can be transferred given the channel's current coin distribution.

### 2.3.2 Rebalancing amounts as network flows

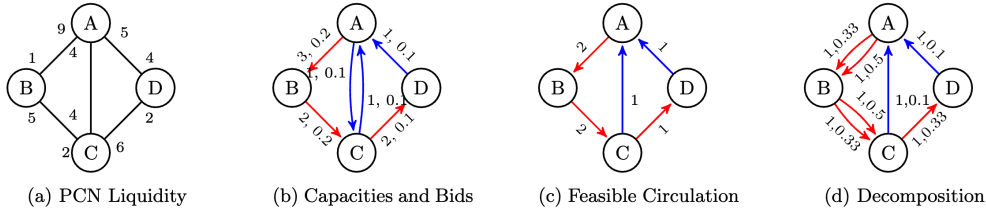
First, all users submit capacities for their channels in both directions. These requests from both buyers and sellers are encoded as a directed capacitated graph  $G = (V, E)$ . For a node  $u$ , the outgoing edges express the channels that  $u$  wishes to send coins to its counterparty – either because the counterparty wishes to rebalance their channel or because  $u$  wants to gain routing fees as a seller. Symmetrically, the incoming edges express the channels that node  $u$  wishes to receive coins – either because  $u$  wishes to rebalance its channel as a buyer or because its counterparty wants to gain routing fees as a seller. We note that it is, therefore, possible to have both directed edges  $(u, v)$  and  $(v, u)$  in  $E$ . The capacities of each edge  $c(e)$ ,  $e \in E$ , represent the maximum amount of flow that the owners of the channel are willing to dedicate to rebalancing. Consequently, the rebalancing problem is now transformed into a network flow problem, e.g., maximizing the rebalancing liquidity is equivalent to maximizing the flow in  $G$ , as illustrated in Figure 3(b).

From the perspective of one user, rebalancing simply transfers their liquidity from one channel to another, possibly depleted, channel. Rebalancing by itself must not result in any monetary gain or loss for any user, a property also known as *balance conservation* [10]. This does not include the fees associated with rebalancing, which may very well lead to a surplus or deficit for users. This requirement of balance conservation characterizes possible

rebalancing flows as circulations. A circulation is a flow  $\mathbf{f} = (f(u, v))_{(u, v) \in E}$  such that the net flow through each vertex is zero:  $\sum_{v \in V} f(u, v) = \sum_{v \in V} f(v, u), \forall u \in V$ .

Two circulations  $\mathbf{f}_1, \mathbf{f}_2$  can be added to get yet another circulation:  $\mathbf{f}_1 + \mathbf{f}_2 = (f_1(u, v) + f_2(u, v))_{(u, v) \in E}$ . A cycle is a sequence of vertices  $v_1, v_2 \dots v_k$  such that  $(v_i, v_{i+1}) \in E, \forall 1 \leq i \leq k-1$  and  $(v_k, v_1) \in E$  as well. We equivalently refer to this cycle as  $(e_1, e_2 \dots e_k)$  where  $e_i = (v_i, v_{i+1}), \forall 1 \leq i \leq k-1$  and  $e_k = (v_k, v_1)$ . We call  $k$  the length of this cycle. A cycle flow  $\mathbf{f}$  of weight  $w$  on cycle  $C$  is a circulation where  $f(e) = w, \forall e \in C$  and  $f(e) = 0$  otherwise (cf. Figure 3(c)).

Although all circulations represent possible rebalancings, rebalancing in practice is executed through cycle flows. First, a so-called *sign-consistent cycle decomposition* of a circulation is computed, and these cycles are individually executed [10]. A sign consistent cycle decomposition of a circulation  $\mathbf{f}$  is a set of cycles  $\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_k$  such that  $\mathbf{f} = \sum_i \mathbf{f}_i$  and all the cycles share the same orientation (cf. Figure 3(d)). To be precise, if two cycles  $\mathbf{f}_i, \mathbf{f}_j$  route non-zero flow through an edge  $(u, v)$ , they do so in the same direction:  $\mathbf{f}_i(u, v) > 0$  and  $\mathbf{f}_j(v, u) > 0$  cannot hold simultaneously. A standard result of network flow theory is that any circulation may be expressed as a sum of at most  $|E|$  sign-consistent cycles [2]. We are only interested in the space of feasible circulations  $\mathbf{f}$  that satisfy every capacity constraint:  $\mathbf{f} \leq \mathbf{c}$ .



**Figure 3** We illustrate the rebalancing process of MUSKETEE: (a) Given a PCN with specific liquidity per channel (indicated by the numbers of each node on each edge), (b) the players may submit capacities and bids (the first number indicated the submitted capacity, the arrow indicated the direction they wish to rebalance, while the second number indicates the fees they are willing to pay). Then, (c) the rebalancing circulation is calculated (the number refer to the number of coins to be transfer and the direction is indicated by the arrow), and (d) subsequently decomposed to sign-consistent cycles which are then priced (the multiple arrows indicate that the flow is divided into multiple cycles; the first number is the number of coins to be transferred and the second the fee to be paid). Depleted edges are shown in red and indifferent edges in blue. All numbers are indicative.

### 2.3.3 User valuations

In a two-party channel, rebalancing is not symmetrically beneficial. We define the utilities resulting from rebalancing flows below.

Associated with each user  $u \in V$  is a valuation function  $\mathbf{v}_u$  on the set of flows in  $G$ . We first assume this valuation to be a linear function of  $\mathbf{f}$ , so that by abuse of notation we may treat  $\mathbf{v}_u$  as a function as well as a vector:  $\mathbf{v}_u(\mathbf{f}) = \mathbf{v}_u \cdot \mathbf{f}$ . For an edge  $e$ , we denote by  $\mathbf{v}_u(e)$  the  $e$ -th coordinate of the vector  $\mathbf{v}_u$ .

If channel  $e = (u, v)$  is depleted in the direction from  $u$  to  $v$ , rebalancing should occur from  $v$  to  $u$ . This would benefit user  $u$ , thus  $u$  has a positive valuation for flow along  $e$ :  $\mathbf{v}_u(e) > 0$ . Any incurred fees are also paid by  $u$ , making  $u$  the buyer in this case. However, if a channel is not depleted, it is termed *indifferent*. Flow in either direction is allowed to aid in rebalancing the network but has a non-positive valuation for the channel owners:  $\mathbf{v}_u(e) \leq 0$ .



Flow from  $u$  to  $v$  requires the authorization of the  $u$ , thus fees earned through this flow are paid to  $u$ , termed the seller.

We assume these valuations are local, meaning that the utility of users is not impacted by the flow along non-adjacent channels:  $\mathbf{v}_u(v, w) = 0$  for distinct users  $u, v, w$ . We further presume each user has a probabilistic knowledge of other users' valuations. Finally, we assume that the utility derived from rebalancing by a unit flow along any channel is bounded, encapsulated by  $\|\mathbf{v}_u\|_\infty < 0.1$ . In other words, no user is willing to pay a fee rate greater than 10%, nor can a user demand greater fees for its indifferent edges. A similar concept is already implemented in the Bitcoin Lightning Network for multi-hop payments (approximately equal to 0.03%). We stress our mechanisms function with any maximum fee rate lower than 100%, and the 10% bound is merely indicative.

### 2.3.4 User bids

Similarly to traditional auctions, user valuations are private and they may submit a different bid. Indeed, we assume all players are rational utility-maximizing agents. We call the bids *valid* when they satisfy the above assumptions on valuations.

In our problem, users submit bids  $\mathbf{b}_u$  for their channels reflecting their self-interests, as shown in Figure 3(b). Buyers submit positive bids while sellers submit negative ones, expressing the maximum/minimum amount of fees they are willing to pay/receive, respectively, per unit flow along their channels during rebalancing. Both users in an indifferent channel may participate as sellers. In depleted channels, however, one party can serve as a buyer while the counterparty is precluded from being a seller to avoid necessitating payment from  $u$  to  $v$  for routing flow. Although we distinguish between buyers and sellers for simplicity, note that users may possess multiple depleted and indifferent channels simultaneously. It is more precise to view each user as a strategic agent with specific utilities derived from their edges.

### 2.3.5 Social welfare and utility functions

Recall that individual user valuations  $\mathbf{v}_u$  are local by assumption and are nonzero only for adjacent directed edges. Let  $\mathbf{v}$  be the aggregate valuation function  $\sum_{u \in V} \mathbf{v}_u$ . Given a feasible circulation  $\mathbf{f}$ , the social welfare generated by  $\mathbf{f}$  under  $\mathbf{v}$  is defined as  $\text{SW}(\mathbf{v}, \mathbf{f}) := \mathbf{v} \cdot \mathbf{f}$ . As usual, user utilities are considered quasi-linear: if  $u$  is charged price  $\mathbf{p}$  for participation in a circulation  $\mathbf{f}$  of valuation  $\mathbf{v}_u(\mathbf{f})$ , then the player's utility is  $\mathbf{u}_u(\mathbf{f}, \mathbf{p}) := \mathbf{v}_u(\mathbf{f}) - \mathbf{p}_u$ . An example of pricing a circulation can be seen in Figure 3(d).

For a vertex  $v$ , we use the subscript “ $-v$ ” to denote the situation where  $v$  is removed from consideration.  $G_{-v}$  refers to the subgraph of  $G$  with  $v$  and all edges adjacent to  $v$  removed.  $\mathbf{v}_{-v}, \mathbf{b}_{-v}$  denote valuation and bid vectors with coordinates for edges adjacent to  $v$  removed. Finally,  $\mathbf{u}_{-v}, \mathbf{p}_{-v}$  denote utilities and prices of all players except  $v$ . These vectors may also be considered as elements of the larger class when it is clear from the context.

### 2.3.6 Rebalancing Game

We define here the rebalancing problem termed MUSKETEER.

► **Definition 1** (MUSKETEER). *Consider a game consisting of  $n$  players, one for each of the vertices of a capacitated directed graph  $G(V, E)$  each with valuation (vector)  $\mathbf{v}_v, v \in V$ . The coordinates correspond to the channels of player  $u$ . A Rebalancing Mechanism  $\mathcal{M} : (G, \mathbf{c}, \mathbf{b}) \mapsto (\mathbf{f}_i, \mathbf{p}_i)_{1 \leq i \leq k}$  receives edge capacities  $\mathbf{c}(e)$  and valid valuations  $\mathbf{b}_v$  as bids from each player.  $\mathcal{M}$  computes a feasible circulation  $\mathbf{f}$  as a sign-consistent cycle decomposition  $\mathbf{f}_1, \dots, \mathbf{f}_k$ .*



330 For each cycle flow  $\mathbf{f}_i$ , it computes a price vector  $\mathbf{p}_i = (\mathbf{p}_i(v))_{(v \in V)}$  that each user must pay.  
 331 Each cycle  $\mathbf{f}_i$  yields a utility of  $\mathbf{u}_v(\mathbf{f}_i, \mathbf{p}_i)$  for player  $v$  as given by  $\mathbf{u}_v(\mathbf{f}_i, \mathbf{p}_i) = \mathbf{v}_v(\mathbf{f}_i) - \mathbf{p}_i(v)$ ,  
 332 and the total utility for player  $v$  is  $\mathbf{u}_v(\mathbf{f}) = \sum_i \mathbf{u}_v(\mathbf{f}_i)$ .

333 The following properties should hold:

- 334 1. **Economic Efficiency:** The cycle decomposition  $\mathbf{f}$  maximizes the social welfare under  
 335 the given bids,  $\mathbf{f} = \arg \max \text{SW}(\mathbf{b}, \mathbf{f})$ .
- 336 2. **Cyclic Budget Balance:** The prices per cycle must sum zero  $\sum_v \mathbf{p}_i(v) = 0$ .
- 337 3. **Individual Rationality:** Any cycle flow  $\mathbf{f}_i$  yields non-negative utility to every truthful  
 338 player,  $\mathbf{u}_u(\mathbf{f}_i) \geq 0$ .
- 339 4. **Truthfulness:** Regardless of other players' actions, the best response for utility-maximizing  
 340 players is to bid truthfully,  $\mathbf{b}_v = \mathbf{v}_v$ .

341 In the context of our problem, *economic efficiency* refers to the maximization of the total flow  
 342 weighted by users' bids, resulting in the most beneficial effect of rebalancing. Additionally,  
 343 this property encompasses the prioritization of channels for rebalancing based on their  
 344 respective bids.

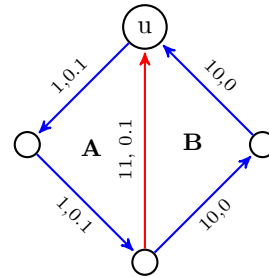
345 *Individual rationality*, on the other hand, demands that each user pays no more than their  
 346 bid for each channel. By adhering to rationality, users ensure that every rebalancing cycle  
 347 yields non-negative utility for all players. Therefore, executing all suggested cycles, rather  
 348 than selectively performing only those that are optimal to their self-interest, is necessary to  
 349 achieve the maximal beneficial effect of rebalancing.

350 *Truthfulness* is another crucial aspect of the mechanism, whereby each player should bid  
 351 their truthful valuation for each channel to ensure that no one can benefit from misreporting  
 352 their valuations.

353 Lastly, *cyclic budget balance* is a novel property tailored to our problem. It is a more  
 354 restrictive variation of the strong budget balance property and demands that there is no  
 355 deficit or surplus for each cycle produced by the mechanism. There are two reasons we  
 356 opt for the cyclic budget balance, both of which stem from PCN technicalities: Firstly,  
 357 the rebalancing circulation is preferably decomposed into cycles. As posited in Hide &  
 358 Seek, executing small cycles is faster, more robust and requires less communication among  
 359 nodes[10]. In contrast, executing the entire circulation simultaneously demands complex  
 360 protocols (such as that of [3]) with high network overhead, which are more likely to fail.  
 361 Secondly, payment channel constructions do not allow the burning of coins, or in other  
 362 words, a surplus for the mechanism. This is because the two users may cooperatively update  
 363 the channel state later in order to split the burned coins, effectively reversing the "burn".  
 364 Therefore, each cycle must be priced in a way that all coins are distributed among the players  
 365 in the cycle. However, we showcase below that attaining cyclic budget balance is strictly  
 366 harder than strong budget balance.

### 367 Hardness of Cyclic Budget Balance

368 Let us demonstrate the increased complexity of at-  
 369 taining cyclic budget balance in comparison to strong  
 370 budget balance (in conjunction with individual ratio-  
 371 nality). The following example (Figure 4) shows that  
 372 the feasible region for strong budget balance exceeds  
 373 that of cyclic budget balance: Suppose player  $u$  sub-  
 374 mits a bid of 0.1 per unit flow for his depleted channel  
 375  $e$  with a rebalancing capacity of 11.  $u$  participates



■ **Figure 4** Depleted edges are depicted with red and indifferent edges with blue. The numbers on each edge indicate the rebalancing capacities and bids, while the rebalancing directions are indicated by the arrows.

in two cycles,  $A$  and  $B$ .  $A$  consists of two indifferent edges bidding  $-0.1$  each (total  $-0.2$  per unit flow) with capacity 1, while  $B$  is composed of two indifferent edges with 0 bids and capacity 10. Regardless of the chosen budget balance property, cycle  $B$  can be selected. However, cycle  $A$  fails to satisfy cyclic budget balance as any rational pricing would result in a deficit of  $-0.1$  per unit flow. However, strong budget-balanced solutions may include both cycles  $A$  and  $B$ , having  $u$  pay  $0.2/11 < 0.1$  fees per unit flow *on average*. Thus, cyclic budget balance restricts the solution space more than strong budget balance.

### 3 Towards Truthful Rebalancing

In this section, we explore how to provide incentive-compatible rebalancing in various settings using auction theory, yielding a flurry of results. In particular, we first prove that satisfying all the desired properties of the MUSKETEER is impossible by applying the classic Myerson-Satterthwaite impossibility result for double auctions (Section 3.1).

To circumvent the impossibility, we present a variety of mechanisms, all of which relax the notion of economic efficiency by restricting the set of possible bids we consider when maximizing social welfare. In particular, we first consider the limited setting where buyers and sellers choose to participate in the mechanism knowing upfront the maximum and minimum fees they would potentially pay or gain, respectively (Section 3.2). The presented algorithm is fairly simple but restricts the choices for participants.

To expand our results to the broader context where players are allowed to submit bids, we relax our model to a single auction, solely considering the buyer's incentives. Specifically, we assume players are willing to forward flow through their indifferent edges hoping to earn some fees in the process, but without a guarantee on the fees. Under this assumption, we present a VCG-type mechanism, satisfying incentive compatibility for buyers (Section 3.3).

Next, we present a double-auction mechanism that takes into account the bids of both buyers and sellers, albeit sacrificing truthfulness, similarly to a first price auction (Section 3.4).

Finally, we leverage time delays to navigate around the impossibility result and design a novel double auction that satisfies all the desiderata in exchange for some costs that users incur in the form of time delays (Section 3.5).

In the following, we provide a high-level description of the various mechanisms, named after the Four Musketeers, highlighting their different design choices and trade-offs. The algorithm facilitating the cycle decomposition is abstracted from the exposition of these mechanisms, and the protocol implementing the atomic execution of these cycles is likewise not detailed. Indicative algorithms that realize these functions can be found in [10] as well as in Section 3.6 for completeness.

## 3.1 Impossibility Result

► **Theorem 2.** *No mechanism can simultaneously satisfy all the desired properties of MUSKETEER, namely economic efficiency, individual rationality, truthfulness, and cyclic budget balance.*

**Proof.** We formulate the double auction problem as a rebalancing game, thus showing that if all properties are satisfied in MUSKETEER then that would be true also for the double auction problem, hence the impossibility of Myerson-Satterthwaite does not hold.

Suppose  $A$  wishes to sell an item and  $B$  wishes to buy it, with individual valuations  $V_a, V_b$  respectively. Each player knows their own valuation with certainty but the valuation of the other player only probabilistically. Without loss of generality, we normalize the valuations to lie in  $[0, 1]$ .

Now construct the following instance of MUSKETEER: the graph  $G = (V, E)$  consists of  $V = \{a, b, c\}$ ,  $E = \{(a, c), (c, b), (b, a)\}$  and with  $\mathbf{c}(e) = 1, \forall e \in E$ . For a flow  $\mathbf{f} = (f_1, f_2, f_3)$  - that is,  $f_1$  units going from  $a$  to  $c$ ,  $f_2$  from  $c$  to  $b$ , and  $f_3$  from  $b$  to  $a$  - set the valuations  $\mathbf{v}_a(\mathbf{f}) = -V_a f_1, \mathbf{v}_b(\mathbf{f}) = V_b f_2, \mathbf{v}_c(\mathbf{f}) = 0$ . We suppose that the players submitted bids  $\mathbf{b}_a, \mathbf{b}_b, \mathbf{b}_c$  respectively, and in particular that  $c$  was honest:  $\mathbf{b}_c = \mathbf{v}_c = 0$ .

The only non-zero feasible circulation is  $\mathbf{f} := (1, 1, 1)$ , so that the mechanism must decide solely between  $\mathbf{f}$  and 0. It must also choose a price vector  $\mathbf{p}$  satisfying cyclic budget balance:  $\mathbf{p}_a + \mathbf{p}_b + \mathbf{p}_c = 0$ .

We interpret choosing  $\mathbf{f}$  as a trade occurring between  $A$  and  $B$ , and choosing 0 as no trade. An efficient mechanism must output  $\mathbf{f}$  if  $V_b > V_a$  (the buyer values the commodity more than the seller). This corresponds to Pareto Efficiency. Individual rationality of players  $a, b$  directly corresponds to individual rationality of  $A$  and  $B$ . Next, individual rationality of  $c$  (the ‘‘auctioneer’’) demands that  $\mathbf{p}_c \leq \mathbf{b}_c = 0$ , which corresponds to Weak Budget Balance. Truthfulness in our setting matches that of Myerson-Satterthwaite: in both cases, we require the truthful bid to be the best response.

In this manner, a solution to MUSKETEER can be used to simulate a single buyer single seller trade as studied by Myerson and Satterthwaite [33]. As a result, all four desired properties cannot be concurrently realized without additional assumptions. ◀

### 3.2 Athos: A Mechanism for Fixed Fees

In this section, we present a straightforward approach for incorporating fees into rebalancing. To circumvent the aforementioned impossibility, the input to the mechanism is restricted. Users do not submit bids. Instead, a predetermined fee rate of  $\hat{p}$  is made publicly known (such as the most commonly chosen fee rate<sup>1</sup>). All flow through indifferent channels will be paid at this fee rate. There is an additional parameter  $k$  that bounds the maximum fee rate for buyers: flow through depleted edges will be charged at a fee rate  $\leq k\hat{p}$ .

Given these parameters, users can decide upfront if they want to participate in the mechanism. Instead of bidding, they specify which of their channels are depleted.  $D \subseteq E$  denotes the set of depleted edges, and the rest are considered indifferent edges, denoted by  $I = E \setminus D$ . The rebalancing flow is chosen to optimize:  $\sum_{e \in D} k\mathbf{f}(e) - \sum_{e \in I} \mathbf{f}(e)$ . The rebalancing is then decomposed into sign-consistent cycles, and a separate price vector is computed for each cycles that achieves cyclic budget balance. This way we achieve all the desiderata but under a restricted setting.

This structure leads to a simple mechanism (ATHOS), a natural evolution of Hide & Seek including fees. We observe however that certain rebalancing cycles are not considered: given the parameter  $k$ , any rebalancing cycle must contain at least one depleted edge for every  $k$  indifferent edges.

► **Theorem 3.** *ATHOS:  $(G, \mathbf{c}, D) \mapsto (\mathbf{f}_i, \mathbf{p}_i)_{1 \leq i \leq k}$ ,  $D \subseteq E$  expressing the set of depleted edges, satisfies economic efficiency, individual rationality, and cyclic budget balance. It also provides sellers with a fee of  $\hat{q} \leq k\hat{p}$  per unit flow along their edges.*

<sup>1</sup> Bitcoin Lightning fees: [https://www.reddit.com/r/lightningnetwork/comments/tm1k1kc/bmonthly\\_ln\\_fee\\_report/](https://www.reddit.com/r/lightningnetwork/comments/tm1k1kc/bmonthly_ln_fee_report/)

**ATHOS: Fixed fees**

*Input:* Channel capacities  $\mathbf{c}$  and the set of depleted edges  $D \subseteq E$ .

1. Let  $I = E \setminus D$  be the set of indifferent edges.
2. Compute the optimal rebalancing  $\mathbf{f} := \arg \max_G \sum_{e \in D} \hat{p} \mathbf{f}(e) - \sum_{e \in I} \hat{p} \mathbf{f}(e)$ .
3. For this flow, the total cost incurred is  $C = \hat{p} \sum_{e \in I} \mathbf{f}(e)$ .
4. Consider a sign-consistent cycle decomposition  $\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_k$  of  $\mathbf{f}$ , and define the cost incurred per cycle as  $C_i = \hat{p} \sum_{e \in I} \mathbf{f}_i(e)$ .
5.  $C_i$  is distributed to the depleted edges in  $\mathbf{f}_i$ . Notice that every cycle  $\mathbf{f}_i$  must contain at least one depleted edge per  $k - 1$  indifferent edges, otherwise remove  $\mathbf{f}_i$  from  $\mathbf{f}^*$  to get a more optimal solution.
6. If the  $i$ th cycle  $\mathbf{f}_i$  contains  $n_i$  depleted edges, then each depleted edge is charged at fee rate  $C_i/n_i$  during the execution of  $\mathbf{f}_i$ . All indifferent edges earn fees at rate  $\hat{p}$ .

*Output:* Cycle flows with prices  $(\mathbf{f}_i, \mathbf{p}_i)$ , each released only to involved players.

**Proof.** This mechanism assumes bids of  $k\hat{p}, \hat{p}$  for depleted and indifferent edges resp., and selects a circulation maximizing social welfare under these bids, thus achieving economic efficiency.

Step 3 clearly indicates that sellers receive a fixed fee for each unit of flow. The parameters  $\hat{p}, k$  are publicly known in advance, hence a user can decide a priori whether it is beneficial to participate in ATHOS based on their private valuations. Individual rationality of player is thus implicit in their participation in the mechanism, along with the fact that all indifferent edges earn fees at rate  $\hat{p}$ , and depleted edges are charged at rate  $\leq k\hat{p}$ .

The fee computation in Step 4 is cyclic budget balanced by design: since we consider the cycle decomposition of  $\mathbf{f}^*$  and charge fees per cycle, the fees charged to depleted edges are identical to the fees levied by the indifferent edges. ◀

### 3.3 Porthos: A Truthful Single Auction

The impossibility of Section 3.1 indicates that achieving all the desiderata is not possible for both buyers and sellers in the original setting. In particular, in our setting, the cyclic budget balance property is critical since burning coins is not possible in payment channels. For this reason, the most straightforward way to circumvent the aforementioned impossibility is to either restrict our setting, as in Section 3.2 where the bids were fixed and known a priori, or revert to a single auction by assuming that sellers will accept any reward that is non-negative.

We, thereby, present here a single auction mechanism where only non-negative bids are permitted: positive bids for depleted channels, and zero for indifferent channels. Instead, all the users of a PCN may participate in the rebalancing process hoping to receive some fees from the mechanism.

We construct a VCG-type mechanism to determine the price vector of buyers based on their impact on social welfare, achieving incentive compatibility for buyers. Charging these prices would result in some surplus for the mechanism, which is instead redistributed to owners of indifferent channels to achieve cyclic budget balance.

**PORTHOS: A VCG-type single auction**

*Input:* Channel capacities  $\mathbf{c}$  and non-negative player bids  $\mathbf{b}_v \geq 0$ .

1. Compute the optimal rebalancing  $\mathbf{f} := \arg \max_G \text{SW}(\mathbf{b}, \mathbf{f})$ .
2. Compute an alternative rebalancing for every player  $v$ ,  $\mathbf{f}_{-v} := \arg \max_G \text{SW}(\mathbf{b}_{-v}, \mathbf{f})$ .
3. Charge  $v$  the price  $\mathbf{p}(v) := \text{SW}(\mathbf{b}_{-v}, \mathbf{f}_{-v}) - \text{SW}(\mathbf{b}_{-v}, \mathbf{f})$ .
4. Let  $\mathbf{f}_1, \dots, \mathbf{f}_k$  be a sign-consistent cycle decomposition of  $\mathbf{f}$ . Non-zero prices  $\mathbf{p}(v)$  are split into  $\mathbf{p}_i(v)$  for each  $\mathbf{f}_i$  proportional to  $v$ 's valuation of  $\mathbf{f}_i$ :  $\mathbf{p}_i(v) := \mathbf{p}(v) \frac{\text{SW}(\mathbf{b}_v, \mathbf{f}_i)}{\text{SW}(\mathbf{b}_v, \mathbf{f})}$ .
5. The total fees per cycle  $\mathbf{f}_i$  are  $q_i = \sum \mathbf{p}_i(v)$  for every buyer in  $\mathbf{f}_i$ .
6. If  $\mathbf{f}_i$  has  $m$  sellers  $u_1, u_2, \dots, u_m$ , then  $p_i(u_j) := -\frac{\sum q_i}{m}$ .

*Output:* Cycle flows with prices  $(\mathbf{f}_i, \mathbf{p}_i)$ , each released only to involved players.

► **Theorem 4.** *PORTHOS:  $(G, \mathbf{c}, \mathbf{b}) \mapsto (\mathbf{f}_i, \mathbf{p}_i)_{1 \leq i \leq k}$  assuming  $\mathbf{b} \geq 0$ , satisfies economic efficiency, individual rationality, and cyclic budget balance. Users' bids for depleted edges are truthful.*

**Proof.** A feasible circulation  $\mathbf{f}$  that maximizes social welfare under  $\mathbf{b}$  achieves economic efficiency. For a player  $v$ , let  $\mathbf{f}_{-v}$  be a feasible circulation on  $G$  maximizing social welfare under bids  $\mathbf{b}_{-v}$ . We set  $\mathbf{p}(v) := \text{SW}(\mathbf{b}_{-v}, \mathbf{f}_{-v}) - \text{SW}(\mathbf{b}_{-v}, \mathbf{f})$ .

It is sufficient to show truthfulness under the pricing  $\mathbf{p}'(v) := -\text{SW}(\mathbf{b}_{-v}, \mathbf{f})$ , since  $\mathbf{p}$  and  $\mathbf{p}'$  are revenue equivalent: meaning that their difference  $\mathbf{p} - \mathbf{p}'$  is a function of  $\mathbf{b}_{-v}$  and  $G_{-v}$ , and this function crucially does not depend on player  $v$ 's bid or valuation.

Under  $\mathbf{p}'$ , player  $v$  is incentivized to bid truthfully regardless of every other player's action. Consider  $\mathbf{b} = (\mathbf{v}_v, \mathbf{b}_{-v})$ ,  $\mathbf{b}' = (\mathbf{v}'_v, \mathbf{b}_{-v})$  for any other valuation  $\mathbf{v}'_v \neq \mathbf{v}_v$ . When  $v$  reports valuation honestly, the mechanism selects  $\mathbf{f}$  maximizing social welfare under  $\mathbf{b}$ , and player  $v$ 's utility is given by  $\mathbf{v}_v(\mathbf{f}) - \mathbf{p}'(v) = \mathbf{v}_v(\mathbf{f}) + \text{SW}(\mathbf{b}_{-v}, \mathbf{f}) = \text{SW}(\mathbf{b}, \mathbf{f})$ . In the second case, the mechanism selects a possibly different  $\mathbf{f}'$  maximizing social welfare under  $\mathbf{b}'$  and the utility for player  $v$  is:  $\mathbf{v}_v(\mathbf{f}') + \text{SW}(\mathbf{b}_{-v}, \mathbf{f}') = \text{SW}(\mathbf{b}', \mathbf{f}')$ . Since  $\text{SW}(\mathbf{b}, \mathbf{f}') \leq \text{SW}(\mathbf{b}, \mathbf{f})$  by definition of  $\mathbf{f}$ , we have that bidding honestly always achieves the maximum possible utility regardless of other players' actions. In other words, both pricings  $\mathbf{p}, \mathbf{p}'$  are Nash-equilibrium incentive-compatible.

Finally, we show individual rationality, or that buyer utilities are non-negative under price  $\mathbf{p}$ . Buyer  $v$ 's utility is  $\mathbf{u}_v = \text{SW}(\mathbf{b}, \mathbf{f}) - \text{SW}(\mathbf{b}_{-v}, \mathbf{f}_{-v})$  which must be non-negative as  $\text{SW}(\mathbf{b}_{-v}, \mathbf{f}_{-v}) \leq \text{SW}(\mathbf{b}, \mathbf{f}_{-v}) \leq \text{SW}(\mathbf{b}, \mathbf{f})$  by definition of  $\mathbf{f}$ . We note that the first inequality only holds for non-negative bids  $\mathbf{b}_v$ .

By the computation in Step 5, the fees charged to depleted edges are equally distributed to all indifferent edges for each cycle. In other words, PORTHOS satisfies cyclic budget balance. ◀

### 3.4 Aramis: A non-truthful Double Auction

As a stepping stone to Section 3.5, we present a straightforward double-auction mechanism (Mechanism ARAMIS) that accepts both positive and negative bids, and satisfies all properties but truthfulness. The rationale of Algorithm 3 resembles that of a first-price auction.

► **Theorem 5.** *ARAMIS:  $(G, \mathbf{c}, \mathbf{b}) \mapsto (\mathbf{f}_i, \mathbf{p}_i)_{1 \leq i \leq k}$  satisfies economic efficiency, individual rationality, and cyclic budget balance, but not truthfulness.*

**ARAMIS: A Double Auction**

*Input:* Channel capacities  $\mathbf{c}$  and player bids  $\mathbf{b}_v$ .

1. Compute the optimal rebalancing  $\mathbf{f} := \arg \max_G \text{SW}(\mathbf{b}, \mathbf{f})$ .
2. Let  $\mathbf{f}_1, \dots, \mathbf{f}_k$  be a sign-consistent cycle decomposition of  $\mathbf{f}$ .
3. Suppose  $\mathbf{f}_i$  is a cycle flow of length  $n_i$ . The price  $\mathbf{p}_i(v)$  for  $v$ 's participation in  $\mathbf{f}_i$  is:  

$$\mathbf{p}_i(v) := \mathbf{b}_v(\mathbf{f}_i) - \frac{\text{SW}(\mathbf{b}, \mathbf{f}_i)}{n_i} \quad (\mathbf{p}_i(v) = 0 \text{ when } v \text{ is not part of } \mathbf{f}_i).$$

*Output:* Cycle flows with prices  $(\mathbf{f}_i, \mathbf{p}_i)$ , each released only to involved players.

**Proof.** The feasible circulation  $\mathbf{f}$  maximizes social welfare under  $\mathbf{b}$  and thus achieves economic efficiency. The social welfare of each cycle  $\mathbf{f}_i$  under  $\mathbf{b}$  must be non-negative, else the circulation  $\mathbf{f} - \mathbf{f}_i$  would have greater social welfare than  $\mathbf{f}$ , contradicting its optimality. Intuitively, the social welfare per cycle is shared uniformly by all involved vertices.

For a truthful player  $v$ , their utility under a cycle  $\mathbf{f}_i$  of length  $n_i$  is given by  $\mathbf{u}_{i,v}(\mathbf{f}_i) = \mathbf{v}_{i,v}(\mathbf{f}_i) - \mathbf{p}_{i,v}(\mathbf{f}_i) = \frac{\text{SW}(\mathbf{b}, \mathbf{f}_i)}{n_i} \geq 0$ . This proves individual rationality per cycle. From the price calculation in Step 3, we can readily confirm that the sum of the prices along each cycle is zero:

$$\sum_{j=1}^{n_i} \mathbf{p}_{i,v_j}(\mathbf{f}_i) := \sum_{j=1}^{n_i} \mathbf{b}_{v_j}(\mathbf{f}_i) - \text{SW}(\mathbf{b}, \mathbf{f}_i) = 0. \quad \blacktriangleleft$$

**Remark.** Players' incentives mirror first-price auctions: They are incentivized to bid higher to ensure their participation in the rebalancing circulation over other competing players. But for a given rebalancing circulation, players are incentivized to bid lower to maximize utility.

### 3.5 d'Artagnan: A Truthful Double Auction with Time Delays

Mechanism ARAMIS is straightforward but lacks the crucial property of truthfulness. To mitigate this issue, we introduce time delays into the rebalancing cycles (mechanism D'ARTAGNAN). The basic concept is that cycles with lower social welfare will be released later in time. Consequently, users who attempt to save on fees by underbidding will experience an undesirable delay in rebalancing. This concept is akin to that of opportunity cost, where users face potential losses from the inability to use their locked funds.

D'ARTAGNAN first computes an optimal rebalancing circulation and decomposes into sign-consistent cycle flows with prices  $(\mathbf{f}_i, \mathbf{p}_i)$ , similar to ARAMIS. Moreover, D'ARTAGNAN selects a time  $t_i \in [0, 1]$  for every flow  $\mathbf{f}_i$  (e.g.,  $t = 1$  represents an 1 hour delay). We assume delaying execution until time  $t_i \leq 1$  gives player  $v$  a utility of  $\mathbf{u}_v = \mathbf{v}_v(\mathbf{f}) - \mathbf{p}_v + d(1 - t)$ . Users join the mechanism with the implicit assumption that rebalancing cycles are released at time  $t = 1$ . Any earlier rebalancing improves the utility of a player at the rate  $d$ , a configurable parameter of our mechanism that depicts the estimated opportunity costs of players.

► **Theorem 6.** *D'ARTAGNAN:  $(G, \mathbf{c}, \mathbf{b}, d) \mapsto (\mathbf{f}_i, \mathbf{p}_i)_{1 \leq i \leq k}$  where  $d$  is an additional delay parameter, satisfies economic efficiency, truthfulness, cyclic budget balance, and individual rationality.*

**Proof.** Cyclic budget balance and economic efficiency follow as in Mechanism PORTHOS since Steps 1 – 3 are identical in both PORTHOS and D'ARTAGNAN. To analyze individual

**D'ARTAGNAN: A Double Auction with delays**

*Input:* Channel capacities  $\mathbf{c}$ , player bids  $\mathbf{b}_v$ , and global *delay factor*  $d$ .

1. Compute the optimal rebalancing  $\mathbf{f} := \arg \max_G \text{SW}(\mathbf{b}, \mathbf{f})$ .
2. Let  $\mathbf{f}_1, \dots, \mathbf{f}_k$  be a sign-consistent cycle decomposition of  $\mathbf{f}$ .
3. Suppose  $\mathbf{f}_i$  is a cycle flow of length  $n_i$ . The price  $\mathbf{p}_i(v)$  for  $v$ 's participation in  $\mathbf{f}_i$  is:  

$$\mathbf{p}_i(v) := \mathbf{b}_v(\mathbf{f}_i) - \frac{\text{SW}(\mathbf{b}, \mathbf{f}_i)}{n_i}.$$
 $\mathbf{p}_i(v)$  is set to zero when  $v$  is not part of the cycle flow.
4. Let  $n_i$  be the length of the cycle flow  $\mathbf{f}_i$ . Define the delay of  $\mathbf{f}_i$  as  

$$t_i = 1 - \left(1 - \frac{1}{n_i}\right) \frac{\text{SW}(\mathbf{b}, \mathbf{f}_i)}{d}.$$

*Output:* The  $i$ th pair  $(\mathbf{f}_i, \mathbf{p}_i)$  is released to involved players at time  $t_i$ .

rationality and truthfulness, let us compute the utility of a player  $v$ . Due to the sign consistency of cycles,  $v$ 's utility can be expressed as the sum of utilities induced by each of the  $k$  cycles:  $\mathbf{u}_v = \sum_i \mathbf{u}_v(\mathbf{f}_i)$ .

The utility of  $v$  per cycle  $\mathbf{f}_i$  is:

$$\begin{aligned} \mathbf{u}_v(\mathbf{f}_i) &= \mathbf{v}_v(\mathbf{f}_i) - \left( \mathbf{b}_v(\mathbf{f}_i) - \frac{\text{SW}(\mathbf{b}, \mathbf{f}_i)}{n_i} \right) + d - dt_i \\ &= \mathbf{v}_v - \mathbf{b}_v + \frac{\text{SW}(\mathbf{b}_v, \mathbf{f}_i)}{n_i} + \left( 1 - \frac{1}{n_i} \right) \text{SW}(\mathbf{b}, \mathbf{f}_i) = (\mathbf{v}_v, \mathbf{b}_{-v}) \cdot \mathbf{f}_i \end{aligned}$$

simplifying to  $\text{SW}((\mathbf{v}_v, \mathbf{b}_{-v}), \mathbf{f}_i)$ . Since  $v$ 's utility is independent of their bid, D'ARTAGNAN is truthful.

In fact, this utility matches the social welfare if bids were honest:  $\mathbf{u}_i(\mathbf{f}_i) = \text{SW}(\mathbf{b}, \mathbf{f}_i)$ . The social welfare of  $\mathbf{f}_i$  cannot be negative. If  $\text{SW}(\mathbf{b}, \mathbf{f}_i) < 0$ , then  $\mathbf{f}$  is not an optimal solution: as  $\mathbf{f}_1, \dots, \mathbf{f}_k$  is a sign consistent cycle decomposition, removal of  $\mathbf{f}_i$  from the circulation  $\mathbf{f}$  leads to a feasible solution that is strictly better. This proves individual rationality. ◀

**Remark.** To guarantee both truthfulness and individual rationality, the users lock their coins to the mechanism a priori for the maximum time delay. Otherwise, buyers may benefit from participating in the mechanism even when the maximum time delay supersedes their true valuations: buyers might only participate in the execution phase (i.e., the sale) if they are quoted a favorable price. This is undesirable behavior as it affects all other users in the cycle. Hence, we enforce the execution of cycles according to the mechanism [38]. However, this hinders economic efficiency, as there may be buyers with ex-ante utility (i.e., utility of player before the output of the mechanism is known) less than their ex-post utility (i.e., utility of player after the output of the mechanism is known). As a result, there may be buyers who would have participated in the mechanism but chose not to, therefore leading to suboptimal outcomes.

### 3.6 Additional Algorithms

In the following, we present for completeness indicative protocols that can implement the cycle decomposition and the atomic execution of these cycles.



579 **Sign-Consistent Cycle Decomposition**

580 We first outline the algorithm for the cycle decomposition, as introduced in [10]. Algorithm 1  
 581 leverages depth-first search to identify cycles and then applies cycle flows to them.

■ **Algorithm 1** Depth-first Search Cycle Decomposition

---

**input** : Circulation  $\mathbf{f}$  on directed graph  $G = (V, E)$   
**output** : A set of cycle flows  $\mathcal{S}$  that sum to  $\mathbf{f}$   
 initialize  $i = 1$   
 initialize  $R \leftarrow \{e \in E : f(e) \neq 0\}$  set of active edges  
**while**  $R \neq \emptyset$  **do**  
   pick an edge  $e_1 \in R$   
   run depth first search to find a cycle  $C_i = (e_1, e_2, \dots, e_k)$  in  $R$   
    $w_i \leftarrow \min f(e), e \in C_i$   
   initialize  $\mathbf{f}_i \leftarrow 0$   
   **for**  $e \in C_i$  **do**  
      $f_i(e) = w_i$   
      $f(e) \leftarrow f(e) - f_i(e)$   
     **if**  $f(e) = 0$  **then**  
        $\perp$  delete  $e$  from  $R$   
    $i \leftarrow i + 1$   
**return**  $\mathcal{S} = \{\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_i\}$

---

582 **Atomic Execution of Rebalancing Cycles**

583 Next, we present an algorithm that ensures the secure atomic execution of the rebalanc-  
 584 ing cycles, taking place after the output of each respective rebalancing mechanism, e.g.  
 585 D'ARTAGNAN.

586 Provided a set of (sign-consistent) rebalancing cycles, Algorithm 2 randomly selects one  
 587 user for each cycle responsible for initiating the execution. This user selects a random number  
 588  $r_c$  and sends its cryptographic digest  $h_c = H(r_c)$  to the other users in its cycle. The initiator  
 589 and the next user have their timelock set to the cycle's length, while the transaction value is  
 590 the cycle's weight  $w_c$ . Each user in the sequence reduces the timelock by 1, identifies the  
 591 next user in the cycle for HTLC creation based on vertex order, and sets up an HTLC with  
 592 the updated timelock.

■ **Algorithm 2** HTLC creation for cycles

---

**input** :  $\mathcal{S}$  set of directed cycles  
**for**  $c \in \mathcal{S}$  **do**  
   select starting user  $u_c$  at random from users in  $c$   
   timelock  $t_c \leftarrow \text{len}(c)$   
    $u_c$  chooses random secret  $r_c$  and creates hash  $h_c = H(r_c)$   
   **for**  $e_c = (u, v) \in c$  *starting from*  $u_c$  **do**  
      $u$  creates  $\text{HTLC}(u, v, w_c, h_c, t_c)$   
      $\perp$  decrement  $t_c$  by 1

---

593 Algorithm 2 follows [10] and is only indicative. It can be replaced by any other protocol  
 594 that achieves atomic execution of multi-hop payments in PCNs, e.g., [49, 47, 46]. For

example, MAPPCN [46] can be leveraged to preserve user anonymity, while MAD-HTLC [47] or He-HTLC [49] can be employed to ensure security even when the blockchain miners can be bribed to enable fraud (so-called timelock bribing attacks [34]).

## 4 Limitations, Extension, and Future Work

Our work leaves open several interesting research avenues which we outline below.

### 4.1 Minimum Fees for Sellers in Aramis

The primary limitation of PORTHOS is that buyer prices rely on the graph structure, resulting in seller's fees being contingent on the number of possible rebalancing cycles in the graph, e.g., if the graph has only one feasible cycle, sellers earn no fees.

A key question is whether it is feasible to guarantee a minimum fee per unit flow through indifferent edges in the mechanism. For a seller, rebalancing is comparable to a typical transaction in the PCN, wherein the seller forwards coins through their channels and earns service fees. Thus, a seller's earnings generally rely on their highly connected position in the network and the amount of capital they have invested. The fee per unit flow (i.e., transfer of one coin) is determined by the intermediary, i.e., the node that sends the coin to the counterparty in their channel. As mentioned earlier, most intermediaries select the same fee per unit flow for forwarding transactions. We thus inquire whether it is feasible to design a novel VCG-style mechanism based on Mechanism PORTHOS, where the graph is modified to guarantee a sufficiently large surplus. Note that the fee earned by sellers is essentially a redistribution of the surplus, and a surplus that is large enough guarantees a minimum fee for every seller.

### 4.2 Incentives

The binary classification of truthfulness is an oversimplification. Future research could aim to quantify and lower bound the potential benefits of misrepresenting bids, such as the gains achieved by underbidding a certain amount instead of truthfully reporting one's valuation.

### 4.3 Variable Delay Costs

In our primary mechanism D'ARTAGNAN, we assume a uniform time delay factor for all players. However, this assumption may not be realistic since different players may experience time delays differently, leading to distinct levels of utility loss. Our model can incorporate this variation by allowing for different delay factors ( $d$ ) for each player. The delay factor can also be construed as the opportunity cost of unused capital in depleted edges, i.e., the potential gain from fees had the player rebalanced his channel. We conjecture that this opportunity cost is quantifiable if buyers furnish their proposed fees for routing since these fees are typically determined by evaluating this loss. Therefore, we can expand our model and require all players, including both buyers and sellers, to submit their anticipated fees. Nevertheless, incorporating this alteration into our model is not a straightforward task. Buyers could potentially manipulate their combined bid by taking into account both the maximum time delay and fees they are willing to incur, consequently violating incentive compatibility.

#### 634 4.4 Repeated Games

635 A pertinent inquiry stemming from the repeated utilization of rebalancing in PCNs is whether  
 636 the expected behavior of players would be altered if they were aware that the rebalancing  
 637 mechanism would occur frequently. Specifically, we ask how would the mechanism design be  
 638 impacted if we shift our game to a repeated setting. We hypothesize that if the rebalancing  
 639 game occurs with sufficient frequency, underbidding may be beneficial as the opportunity to  
 640 rebalance would be reduced but not entirely eliminated. Conversely, if the rebalancing game  
 641 is infrequent, players may miss their chance to rebalance. We thus anticipate that integrating  
 642 frequency-dependent utility losses may significantly alter the results of the rebalancing game.

#### 643 4.5 Group Strategy-Proof Mechanisms

644 Both PORTHOS and D'ARTAGNAN are strategy-proof but not group strategy-proof. While a  
 645 single user's misreported bids cannot improve their utility, in certain cases, two users can  
 646 manipulate their bids to jointly increase their utilities. Consider for instance the parties  
 647  $u, v$  of a depleted channel in PORTHOS. If the channel is depleted from  $u$  to  $v$ , then an  
 648 honest  $u$  would truthfully report a positive bid from  $v$  to  $u$ , thus prohibiting  $v$  from gaining  
 649 routing fees for the  $u, v$  channel. However, both  $u$  and  $v$  may gain by  $u$  misreporting a zero  
 650 bid for the channel. This misrepresentation converts the channel's status from depleted to  
 651 indifferent, enabling the potential for  $v$  to gain routing fees while precluding the possibility  
 652 that  $u$  pays any fees. Given this example, an intriguing open problem is designing group  
 653 strategy-proof mechanisms specifically tailored to counter collusion between a channel's joint  
 654 owners.

## 655 5 Additional Related Work

### 656 5.1 Blockchain Scalability & Payment Channel Networks

657 Improving the blockchain transaction throughput has garnered interest since the inception of  
 658 Bitcoin [35]. Proposed solutions include increasing the block size, sharding the blockchain,  
 659 or moving the workload off-chain leveraging so-called layer-2 protocols such as sidechains,  
 660 channels, and rollups (see [25, 24] for recent surveys). Among these solutions, payment  
 661 channel networks, such as the Bitcoin Lightning Network [38], have attracted substantial  
 662 attention because they enable instant, low-cost off-chain transactions.

663 A large body of research has emerged focusing on various aspects of PCNs, such as  
 664 efficient and privacy-preserving routing, e.g. [43, 39, 37, 31, 4, 45], and algorithmic analysis  
 665 of the PCN topology [6, 5]. In the intersection of PCNs and game theory, there are several  
 666 works, mainly focusing on network topology leveraging network creation games [19, 7, 5],  
 667 and incentive-compatible outsourcing of channels' dispute resolution [32, 9, 8, 30]. All these  
 668 works are orthogonal and complementary to ours as they ignore channel depletion.

669 Perhaps the most relevant work to ours is Merchant [48], employing fee functions as a  
 670 mechanism to avert channel depletion by guiding routing paths. By allowing intermediaries  
 671 to impose varying fees for distinct routes, users are incentivized to prefer specific routes over  
 672 others, ultimately mitigating channel depletion. This method presents a complementary  
 673 approach to our work, in which we propose an opt-in rebalancing protocol to address channel  
 674 depletion.

675 The problem of channel rebalancing has been studied in several recent works [26, 10, 1],  
 676 which we build upon and extend. Our work is the first to consider user incentives in the  
 677 context of rebalancing mechanisms for PCNs.

## 5.2 Game-theoretic Analysis of Blockchains

Numerous studies have investigated incentives in the context of the consensus layer of blockchains. For instance, Pass and Shi introduced an innovative incentive-compatible consensus protocol called FruitChains [36]. Additionally, several works focused on a rational analysis of Bitcoin’s consensus: exploring when rational miners follow the protocol [13, 27], devising attacks that showcase Bitcoin is not incentive-compatible, e.g., [21, 29, 41, 44], investigating the impact of block rewards and mining pools, e.g., [16, 15, 20, 42]. On the other hand, Babaioff et al. [12] explored the network layer of blockchains and proposed an incentive-compatible scheme for information propagation within Bitcoin’s peer-to-peer network. However, these works address different issues from ours, as they focus on the consensus and network layers of blockchains, while our research investigates incentives on layer-2 networks that build upon the other layers.

## 5.3 Mechanism Design on Networks

The rebalancing problem fits into the well-established research area of mechanism design on networks, with an impact on computer science, economics, and operations research. The most relevant examples to our problem include Stackelberg routing, selfish routing in capacitated networks, and optimal oblivious routing, e.g., [17, 11, 40, 28]. Our work differs from the existing literature in several ways. First, we focus on a novel problem domain – rebalancing mechanisms for PCNs – which has not been previously studied from a game-theoretic perspective. Second, we deal with a unique set of constraints due to the nature of payment channels and PCNs, such as channel depletion and cyclic budget balance. These constraints lead to novel challenges in the design of incentive-compatible mechanisms, addressed in this work.

## 6 Conclusion

In this paper, we revisited the challenge of rebalancing in payment channel networks (PCNs) from a mechanism design perspective, introducing a novel approach that takes into account users’ incentives. By incorporating both buyers and sellers of channel liquidity in our proposed rebalancing mechanism, we introduced the double-auction rebalancing problem MUSKETEEER, which aims to optimize the throughput in PCNs.

Our work demonstrates that the unique characteristics of PCNs, particularly the cyclic budget balance property, pose significant challenges in designing a mechanism that simultaneously satisfies all the desiderata. Our results, grounded in auction theory, revealed an impossibility result, leading us to develop a variety of mechanisms that balance the various desiderata. Notably, we introduced a novel mechanism that employs time delays to overcome the impossibility result, successfully meeting all desired properties, albeit at the expense of economic efficiency in terms of time delays and liquidity combined.

## References

- 1 Rebalance plugin. <https://github.com/lightningd/plugins/tree/master/rebalance>.
- 2 R. Ahuja, T. Magnanti, and J. Orlin. Network flows - theory, algorithms and applications. 1993.
- 3 Lukas Aumayr, Kasra Abbaszadeh, and Matteo Maffei. Thora: Atomic and privacy-preserving multi-channel updates. *IACR Cryptol. ePrint Arch.*, page 317, 2022. URL: <https://eprint.iacr.org/2022/317>.

- 721    **4**    Lukas Aumayr, Pedro Moreno-Sanchez, Aniket Kate, and Matteo Maffei. Blitz: Secure  
722       Multi-Hop payments without Two-Phase commits. In *30th USENIX Security Symposium*  
723       (*USENIX Security 21*), pages 4043–4060. USENIX Association, August 2021. URL: <https://www.usenix.org/conference/usenixsecurity21/presentation/aumayr>.  
724
- 725    **5**    Georgia Avarikioti, Gerrit Janssen, Yuyi Wang, and Roger Wattenhofer. Payment network  
726       design with fees. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*,  
727       pages 76–84. Springer, 2018.
- 728    **6**    Georgia Avarikioti, Yuyi Wang, and Roger Wattenhofer. Algorithmic channel design. In *29th*  
729       *International Symposium on Algorithms and Computation, Jiaoxi, Yilan County, Taiwan*,  
730       2018.
- 731    **7**    Zeta Avarikioti, Lioba Heimbach, Yuyi Wang, and Roger Wattenhofer. Ride the lightning:  
732       The game theory of payment channels. In *International Conference on Financial Cryptography*  
733       *and Data Security*, 2020.
- 734    **8**    Zeta Avarikioti, Eleftherios Kokoris Kogias, Roger Wattenhofer, and Dionysis Zindros. Brick:  
735       Asynchronous incentive-compatible payment channels. In *International Conference on Finan-*  
736       *cial Cryptography and Data Security*, 2021.
- 737    **9**    Zeta Avarikioti, Orfeas Stefanos Thyfronitis Litos, and Roger Wattenhofer. Cerberus channels:  
738       Incentivizing watchtowers for bitcoin. In *International Conference on Financial Cryptography*  
739       *and Data Security*, pages 346–366. Springer, 2020.
- 740    **10**   Zeta Avarikioti, Krzysztof Pietrzak, Iosif Salem, Stefan Schmid, Samarth Tiwari, and Michelle  
741       Yeo. Hide and seek: Privacy-preserving rebalancing on payment channel networks. In *Proc.*  
742       *Financial Cryptography and Data Security (FC)*, 2022.
- 743    **11**   Yossi Azar, Edith Cohen, Amos Fiat, Haim Kaplan, and Harald Racke. Optimal oblivious  
744       routing in polynomial time. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on*  
745       *Theory of Computing*, STOC ’03, page 383–388, New York, NY, USA, 2003. Association for  
746       Computing Machinery. doi:10.1145/780542.780599.
- 747    **12**   Moshe Babaioff, Shahar Dobzinski, Sigal Oren, and Aviv Zohar. On bitcoin and red balloons.  
748       In *EC*, 2012. doi:10.1145/2229012.2229022.
- 749    **13**   Christian Badertscher, Juan Garay, Ueli Maurer, Daniel Tschudi, and Vassilis Zikas. But  
750       why does it work? a rational protocol design treatment of bitcoin. In *Eurocrypt*, 2018.  
751       doi:10.1007/978-3-319-78375-8\_2.
- 752    **14**   Burak Can, Jens Leth Hougaard, and Mohsen Pourpouneh. On reward sharing in blockchain  
753       mining pools. *Games and Economic Behavior*, 136:274–298, 2022. URL: <https://www.sciencedirect.com/science/article/pii/S0899825622001361>, doi:10.1016/j.geb.2022.  
754       10.002.  
755
- 756    **15**   Miles Carlsten, Harry Kalodner, S Matthew Weinberg, and Arvind Narayanan. On the  
757       instability of bitcoin without the block reward. In *CCS*, 2016. doi:10.1145/2976749.2978408.
- 758    **16**   Xi Chen, Christos Papadimitriou, and Tim Roughgarden. An axiomatic approach to block  
759       rewards. In *AFT*, 2019. doi:10.1145/3318041.3355470.
- 760    **17**   José R Correa, Andreas S Schulz, and Nicolás E Stier-Moses. Selfish routing in capacitated  
761       networks. *Mathematics of Operations Research*, 33(4):961–976, 2008.
- 762    **18**   Kyle Croman, Christian Decker, Ittay Eyal, Adem Efe Gencer, Ari Juels, Ahmed Kosba, An-  
763       drew Miller, Prateek Saxena, Elaine Shi, Emin Gün Sirer, Dawn Song, and Roger Wattenhofer.  
764       On scaling decentralized blockchains. In *International Conference on Financial Cryptography*  
765       *and Data Security*, pages 106–125. Springer, 2016.
- 766    **19**   Oğuzhan Ersoy, Stefanie Roos, and Zekeriya Erkin. How to profit from payments channels. In  
767       *FC*, 2020. doi:10.1007/978-3-030-51280-4\_16.
- 768    **20**   Ittay Eyal. The miner’s dilemma. In *IEEE S&P*, 2015. doi:10.1109/SP.2015.13.
- 769    **21**   Ittay Eyal and Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. In  
770       *International conference on financial cryptography and data security*, pages 436–454. Springer,  
771       2014.

- 772 22 Joan Feigenbaum, Christos H Papadimitriou, Rahul Sami, and Scott Shenker. A bgp-based  
773 mechanism for lowest-cost routing. *Distributed Computing*, 18(1):61–72, 2005.
- 774 23 Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand:  
775 Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th Symposium on*  
776 *Operating Systems Principles*, pages 51–68. ACM, 2017.
- 777 24 Lewis Gudgeon, Pedro Moreno-Sanchez, Stefanie Roos, Patrick McCorry, and Arthur Gervais.  
778 Sok: Layer-two blockchain protocols. In *International Conference on Financial Cryptography*  
779 *and Data Security*, pages 201–226. Springer, 2020.
- 780 25 Abdelatif Hafid, Abdelhakim Senhaji Hafid, and Mustapha Samih. Scaling blockchains: A  
781 comprehensive survey. *IEEE Access*, 8:125244–125262, 2020. doi:10.1109/ACCESS.2020.  
782 3007251.
- 783 26 Rami Khalil and Arthur Gervais. Revive: Rebalancing off-blockchain payment networks. In  
784 *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*,  
785 pages 439–453, 2017.
- 786 27 Aggelos Kiayias, Elias Koutsoupias, Maria Kyropoulou, and Yiannis Tselekounis. Blockchain  
787 mining games. In *EC*, 2016. doi:10.1145/2940716.2940773.
- 788 28 Y.A. Korilis, A.A. Lazar, and A. Orda. Achieving network optima using stackelberg routing  
789 strategies. *IEEE/ACM Transactions on Networking*, 5(1):161–173, 1997. doi:10.1109/90.  
790 554730.
- 791 29 Yujin Kwon, Dohyun Kim, Yunmok Son, Eugene Vasserman, and Yongdae Kim. Be selfish  
792 and avoid dilemmas: Fork after withholding (faw) attacks on bitcoin. In *CCS*, 2017. doi:  
793 10.1145/3133956.3134019.
- 794 30 Joshua Lind, Oded Naor, Ittay Eyal, Florian Kelbert, Emin Gün Sirer, and Peter R. Pietzuch.  
795 Teechain: a secure payment network with asynchronous blockchain access. In *Proceedings of*  
796 *the 27th ACM Symposium on Operating Systems Principles*, pages 63–79, 2019.
- 797 31 Giulio Malavolta, Pedro Moreno-Sanchez, Aniket Kate, and Matteo Maffei. Silentwhispers:  
798 Enforcing security and privacy in decentralized credit networks. In *24th Annual Network and*  
799 *Distributed System Security Symposium*, 2017.
- 800 32 Patrick McCorry, Surya Bakshi, Iddo Bentov, Sarah Meiklejohn, and Andrew Miller. Pisa:  
801 Arbitration outsourcing for state channels. In *Proceedings of the 1st ACM Conference on*  
802 *Advances in Financial Technologies*, pages 16–30. ACM, 2019.
- 803 33 Roger B Myerson and Mark A Satterthwaite. Efficient mechanisms for bilateral trading.  
804 *Journal of Economic Theory*, 29(2):265–281, 1983. URL: <https://www.sciencedirect.com/science/article/pii/0022053183900480>, doi:10.1016/0022-0531(83)90048-0.
- 805 34 Tejaswi Nadahalli, Majid Khabbazi, and Roger Wattenhofer. Timelocked bribing. In  
806 *Financial Cryptography and Data Security*, Lecture Notes in Computer Science, 2021.
- 807 35 Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.
- 808 36 Rafael Pass and Elaine Shi. Fruitchains: A fair blockchain. In *PODC*, 2017. doi:10.1145/  
809 3087801.3087809.
- 810 37 Krzysztof Pietrzak, Iosif Salem, Stefan Schmid, and Michelle Yeo. Lightpir: Privacy-  
811 preserving route discovery for payment channel networks. In Zheng Yan, Gareth Tyson,  
812 and Dimitrios Koutsounikolas, editors, *IFIP Networking Conference, IFIP Networking 2021,*  
813 *Espoo and Helsinki, Finland, June 21-24, 2021*, pages 1–9. IEEE, 2021. doi:10.23919/  
814 IFIPNetworking52078.2021.9472205.
- 815 38 Joseph Poon and Thaddeus Dryja. The bitcoin lightning network: Scalable off-chain instant  
816 payments, 2015.
- 817 39 Stefanie Roos, Pedro Moreno-Sanchez, Aniket Kate, and Ian Goldberg. Settling payments  
818 fast and private: Efficient decentralized routing for path-based transactions. *arXiv preprint*  
819 *arXiv:1709.05748*, 2017.
- 820 40 Tim Roughgarden. Selfish routing and the price of anarchy. *MIT press*, 2005.
- 821 41 Ayelet Sapirshtein, Yonatan Sompolinsky, and Aviv Zohar. Optimal selfish mining strategies  
822 in bitcoin. In *FC*, 2016. doi:10.1007/978-3-662-54970-4\_30.
- 823

- 824 **42** Okke Schrijvers, Joseph Bonneau, Dan Boneh, and Tim Roughgarden. Incentive compatibility  
825 of bitcoin mining pool reward functions. In *FC*, 2016. doi:10.1007/978-3-662-54970-4\_28.
- 826 **43** Vibhaalakshmi Sivaraman, Shaileshh Bojja Venkatakrisnan, Kathleen Ruan, Parimarjan  
827 Negi, Lei Yang, Radhika Mittal, Giulia Fanti, and Mohammad Alizadeh. High throughput  
828 cryptocurrency routing in payment channel networks. In *17th {USENIX} Symposium on  
829 Networked Systems Design and Implementation ({NSDI} 20)*, pages 777–796, 2020.
- 830 **44** Jason Teutsch, Sanjay Jain, and Prateek Saxena. When cryptocurrencies mine their own  
831 business. In *FC*, 2016. doi:10.1007/978-3-662-54970-4\_29.
- 832 **45** Samarth Tiwari, Michelle Yeo, Zeta Avarikioti, Iosif Salem, Krzysztof Pietrzak, and Stefan  
833 Schmid. Wiser: Increasing throughput in payment channel networks with transaction aggrega-  
834 tion. *CoRR*, abs/2205.11597, 2022. arXiv:2205.11597, doi:10.48550/arXiv.2205.11597.
- 835 **46** Somanath Tripathy and Susil Kumar Mohanty. Mappcn: Multi-hop anonymous and privacy-  
836 preserving payment channel network. In *International Conference on Financial Cryptography  
837 and Data Security*, pages 481–495. Springer, 2020.
- 838 **47** Itay Tsabary, Matan Yechieli, and Ittay Eyal. MAD-HTLC: because HTLC is crazy-cheap to  
839 attack. *IEEE S&P*, 2021. URL: <https://arxiv.org/abs/2006.12031>.
- 840 **48** Yuup Van Engelshoven and Stefanie Roos. The merchant: Avoiding payment channel depletion  
841 through incentives. In *2021 IEEE International Conference on Decentralized Applications and  
842 Infrastructures (DAPPS)*, pages 59–68. IEEE, 2021.
- 843 **49** Sarisht Wadhwa, Jannis Stoeter, Fan Zhang, and Kartik Nayak. He-HTLC: Revis-  
844 iting Incentives in HTLC. In *30th Annual Network and Distributed System Secu-  
845 rity Symposium, NDSS 2023, San Diego, California, USA, February 27 - March 3,  
846 2023*. The Internet Society, 2023. URL: [https://www.ndss-symposium.org/ndss-paper/  
847 he-htlc-revisiting-incentives-in-htlc/](https://www.ndss-symposium.org/ndss-paper/he-htlc-revisiting-incentives-in-htlc/).
- 848 **50** Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum  
849 Project Yellow Paper*, 2014.